Applicants : W. James Allen et al.
Appln. No. : 10/075,319
Page : 2

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently Amended) An electronic control unit (ECU), comprising:

a central processing unit (CPU) coupled to a port which couples the CPU to an external tool, the CPU executing a control algorithm which controls a subsystem coupled to the ECU;

a non-volatile memory bank coupled to the CPU, the non-volatile memory bank storing a plurality of calibration tables and a code set;

a volatile memory coupled to the CPU; and

a state machine in communication with the CPU, wherein the state machine functions to selectively capture information available on an internal bus of the CPU on a cycle-by-cycle basis and store the captured information in the volatile memory, and wherein the state machine is implemented by one of another CPU, discrete gates, a field programmable gate array (FPGA) and a digital signal processor (DSP).

2. (Original) The ECU of claim 1, wherein the captured information includes one of a full trace and limited trace of information on at least one of an address and data bus of the CPU.

3. (Original) The ECU of claim 2, wherein the CPU has separate instruction and data buses.

4. (Original) The ECU of claim 2, wherein the CPU uses the same bus for both instructions and data.

5. (Original) The ECU of claim 1, wherein the state machine performs triggering, data acquisition and circular buffer control of a trace buffer located within the volatile memory.

6. (Original) The ECU of claim 1, wherein the state machine includes a counter.

7. (Original) The ECU of claim 1, further including:

an overlay memory that the state machine configures as a trace buffer which captures valid data on the data bus, wherein the overlay memory receives it address from an address counter that is incremented with each clock cycle that contains one of instruction information and data information.

8. (Original) The ECU of claim 7, wherein the trace buffer only captures operational codes and associated data.

9. (Original) The ECU of claim 7, wherein the overlay memory is divided into a first memory bank and a second memory bank with the address counter providing address information for both of the banks, and wherein the first memory bank stores an address for each cycle and the second memory bank stores one of an associated operational code and application data for each cycle.

10. (Original) The ECU of claim 7, wherein the overlay memory is implemented as a dedicated RAM device.

11. (Original) The ECU of claim 7, wherein the overlay memory is implemented as a non-volatile device.

12. (Original) The ECU of claim 1, further including:

a debug port, wherein the debug port is coupled to the CPU.

13. (Original) The ECU of claim 1, wherein the port is one of a debug interface and an existing serial communication port.

14. (Canceled)

15. (Original) The ECU of claim 1, wherein the CPU, non-volatile memory bank, volatile memory and state machine are implemented within a microcontroller unit (MCU).

16. (Original) The ECU of claim 1, further including:
a watchpoint generator for providing a triggering signal to the state machine.

17. (Original) The ECU of claim 1, further including:
a watchpoint generator for providing triggering signals to the state machine.

18. (Original) The ECU of claim 1, wherein the captured information is confined to a specific list by the state machine.

19. (Original) The ECU of claim 1, wherein the captured information is confined to an address range specified by the state machine.

20. (Original) The ECU of claim 19, wherein the captured information is confined to a series of packets stored in the volatile memory.

21. (Original) The ECU of claim 1, wherein the volatile memory is protected from data loss during normal operating conditions.

22. (Currently Amended) The ECU of claim 1, wherein only [[the]] operational code and data of a CPU cycle are saved in the volatile memory and the state machine saves [[the]] an address of a CPU access in a register when a trigger occurs.

23. (Original) The ECU of claim 1, wherein the state machine captures an address of a CPU access in a specified address range.

24. (Original) The ECU of claim 23, wherein the state machine stores the captured address in a specific register.

25. (Original) The ECU of claim 23, wherein the state machine stores the captured address in the volatile memory.

26. (Original) The ECU of claim 23, wherein the state machine stores the captured addresses in the volatile memory as a series of packets.

27. (Currently Amended) An electronic control unit (ECU), comprising:

a central processing unit (CPU) coupled to a port which couples the CPU to an external tool, the CPU executing a control algorithm which controls a subsystem coupled to the ECU;

a non-volatile memory bank coupled to the CPU, the non-volatile memory bank storing a plurality of calibration tables and a code set;

a volatile memory coupled to the CPU;

an overlay memory coupled to the CPU; and

a state machine in communication with the CPU, wherein the state machine functions to selectively capture information available on an internal bus of the CPU on a cycle-by-cycle basis and store the captured information in the overlay memory, and wherein the state machine configures the overlay memory as a trace buffer which captures valid data on the data bus, where the overlay memory receives it address from

an address counter that is incremented with each clock cycle that contains one of instruction information and data information.

28. (Original) The ECU of claim 27, wherein the captured information includes one of a full trace and a limited trace of information on at least one of an address and data bus of the CPU.

29. (Original) The ECU of claim 27, wherein the CPU has separate instruction and data buses.

30. (Original) The ECU of claim 27, wherein the CPU uses the same bus for both instructions and data.

31. (Original) The ECU of claim 27, wherein the state machine performs triggering, data acquisition and circular buffer control of a trace buffer located within the volatile memory.

32. (Original) The ECU of claim 27, wherein the state machine includes a counter.

33. (Canceled)

34. (Currently Amended) The ECU of claim [[33]] 27, wherein the trace buffer only captures operational codes and associated data.

35. (Currently Amended) The ECU of claim [[33]] 27, wherein the overlay memory is divided into a first memory bank and a second memory bank with the address counter providing address information for both of the banks, and wherein the first memory bank stores an address for each cycle and the second memory bank stores one of an associated operational code and data for each cycle.

36. (Original) The ECU of claim 27, further including:

a debug port, wherein the debug port is coupled to the CPU.

37. (Original) The ECU of claim 27, wherein the port is a debug interface.

38. (Original) The ECU of claim 27, wherein the state machine is implemented by one of another CPU, discrete gates, a field programmable gate array (FPGA) and a digital signal processor (DSP).

39. (Original) The ECU of claim 27, wherein the CPU, non-volatile memory bank, volatile memory, overlay memory and state machine are implemented within a microcontroller unit (MCU).

40. (Original) An automotive microcontroller unit (MCU), comprising:

a central processing unit (CPU) coupled to a port which couples the CPU to an external tool, the CPU executing a control algorithm which controls an automotive subsystem coupled to the MCU;

a non-volatile memory bank coupled to the CPU, the non-volatile memory bank storing a plurality of calibration tables and a code set;

a volatile memory coupled to the CPU;

a state machine in communication with the CPU, wherein the state machine functions to selectively capture information available on an internal bus of the CPU on a cycle-by-cycle basis and store the captured information in the volatile memory, and wherein the state machine is implemented by one of another CPU, discrete gates, a field programmable gate array (FPGA) and a digital signal processor (DSP); and

a debug port coupled to the CPU.